Simpler Testing for Two-page Book Embedding of Partitioned Graphs

Seek-Hee Hong[1] Hiroshi Nagamochi[2]

[1]School of Information Technologies, University of Sydney, seokhee.hong@sydney.edu.au
[2]Department of Applied Mathematics and Physics, Kyoto University, nag@amp.i.kyoto-u.ac.jp

Abstract: A *2-page book embedding* of a graph is to place the vertices linearly on a spine (a line segment) and the edges on the two pages (two half planes sharing the spine) so that each edge is embedded in one of the pages without edge crossings. Testing whether a given graph admits a 2-page book embedding is known to be NP-complete.

In this paper, we study the problem of testing whether a given graph admits a 2-page book embedding with a fixed edge partition. We first show that finding a 2-page book embedding of a given graph can be reduced to the planarity testing of a graph, which yields a simple linear-time algorithm for solving the problem. We also characterize the graphs that do not admit 2-page book embeddings via forbidden subgraphs, and give a linear-time algorithm for detecting the forbidden subgraph of a given graph.

1 Introduction

For an integer $k \geq 1$, a k-page book embedding (or a k-stack layout) of a graph is to place the vertices linearly on a spine (a line segment) and the edges on k pages (k half planes sharing the spine) so that each edge is embedded in one of the pages without generating edge-crossings. The book embedding problem has applications in the routing of multilayer printed circuit boards and in the design of fault-tolerant processor arrays [3, 19]. See [11] for numerous applications of book embeddings.

Graphs with 1-page book embeddings are the outerplanar graphs. Yannakakis proved that every planar graph admits a 4-page book embedding [21]. Bernhart and Kainen [2] show that a planar graph has a 2-page book embedding if and only if it is sub-Hamiltonian. A planar graph is sub-Hamiltonian if and only if it is Hamiltonian or can be made Hamiltonian by inserting additional edges without violating planarity. The problem of testing sub-Hamiltonicity is NP-complete [22]. Hence the problem of determining whether a given planar graph $G = (V, E)$ is a 2-page book embedding or not is NP-complete. See [11, 12] for a survey on book embeddings and graph linear layouts.

[1]Technical report 2013-001, February 12, 2013.

The 2-page book embedding problem contains two combinatorial aspects. One is how to partition an edge set in two edge subsets, each corresponds to one of the two sides along the spine. The other is how to decide an ordering of the vertices on the spine. Note that if an ordering π of all the vertices along the spine is fixed, then we can test whether a given graph admits a 2-page book embedding with π or not in linear time; the problem can be converted into a planarity testing problem by adding edges between every two consecutive vertices in π (where the last vertex is connected by the first one). However, it is not known whether the problem remains NP-complete or can be solved in polynomial time if a partition of the edge set is prescribed.

In this paper, we consider the problem of testing whether a given graph admits a 2-page book embedding for a fixed edge partition. Based on structural properties of biconnected planar graphs, we show that the problem of finding a 2-page book embedding of a graph with a partitioned edge set can be solved in linear time.

A preliminary version of this paper appeared as a report [16], which has been recently used to solve several other graph drawing problems such as simultaneous drawing of two graphs. In [16], we characterize 2-page book embeddings as "splitter-free" and "disjunctive" plane embeddings (see Section 6 for the definitions), and show that such an embedding (if any) can be constructed in linear time by designing three procedures, one for detecting "rigid" splitters, a special type of splitters, one for computing splitter-free plane embeddings and the other for computing disjunctive plane embeddings. Recently Angelini et al. [1] implemented the algorithm in in [16] after they simplified the procedure for computing "disjunctive" plane embeddings reducing the hidden constant factor in the time bound.

In this paper, we first show that a given instance can be converted into another instance with a special structure, called a "canonical" instance (see Section 6 for the definition). We then prove that finding a 2-page book embedding of a given graph can be reduced to the planarity testing of a modified graph, which yields a simple linear-time algorithm for solving the problem without using any of the three procedures in [16]. We also characterize the graphs that do not admit 2-page book embeddings via forbidden subgraphs, and give a linear-time algorithm for detecting the forbidden subgraph of a given graph. This algorithm is obtained by significantly simplifying the first procedure for detecting rigid splitters utilizing the restricted structure of canonical instances.

The paper is organized as follows. In Section 2, we review basic terminologies on graph connectivity and outerplanar graphs In Section 3, we show how to convert a given instance into a canonical instance, an instance with a special structure. In Section 4, we characterize a 2-page book embedding as a plane embedding which satisfies two conditions: it has no splitter and it is disjunctive. In Section 5, we show that the 2-page book embedding of a canonical instance can be reduced to the planarity testing problem on a modified graph. In Section 6, after we review SPQR tree representation of triconnected components, we design a linear-time algorithm for detecting a forbidden subgraph of an infeasible instance.

2 Preliminaries

Let $G = (V, E)$ be a graph. The set of edges incident to a vertex $v \in V$ is denoted by $E(v; G)$. A path with end vertices u and v is called a *u, v-path*. The degree of a vertex v in G is denoted by $\deg(v; G)$. For a subset $X \subseteq E$ (resp., $X \subseteq V$), $G - X$ denotes the graph obtained from G by removing the edges in X (resp., the vertices in X together with the edges in $\cup_{v \in X} E(v; G)$). *Subdividing* an edge $e = (u, v)$ is to replace the edge with a u, v-path $u, w_1, w_2, \ldots, u_k, v$ for some $k \geq 1$. A graph H is a subdivision of G if H is obtained by subdividing some edges in G.

A *block* (biconnected component) of a graph is a maximal biconnected subgraph (which possibly consists of a single vertex or a single edge). A graph each of whose blocks is a simple cycle or a single edge is called a *cactus*, i.e., a graphs in which any two distinct cycles share at most one vertex. We see that a graph is a cactus if and only if no two vertices are joined by three vertex-disjoint paths.

A planar graph $G = (V, E)$ with a fixed embedding F of G is called a *plane* graph. The set of vertices, set of edges and set of facial cycles of a plane graph H may be denoted by $V(H)$, $E(H)$ and $F(H)$, respectively. We say that a cycle Q in a plane embedding of a graph *separates* a vertex/edge a_1 and a vertex/edge a_2 (which are not elements in Q) if a_1 and a_2 are respectively contained in the two regions R_1 and R_2 obtained by dividing the plane by Q.

A planar graph is called *outerplanar* if it admits a plane embedding such that all the vertices appear along the outer boundary, which we call an *outerplane* embedding. We see that each block B of a simple outerplanar graph is a single edge or a cycle Q of length at least 3 possibly with some chords (see Fig. 1(a) for an example of outerplanar graphs). Note that such a cycle Q for the block B is uniquely determined, and we call Q the *frame* of B (we let $Q = B$ if B is a single edge). We observe the next.

Lemma 1 *Let B_i, $i = 1, \ldots, p$ be the blocks of a simple outerplanar graph G, and Q_i be the boundary of B_i. Then a plane embedding Γ_G of G is an outerplane embedding of G if and only if each frame Q_i appears on the outer boundary of Γ_G.*

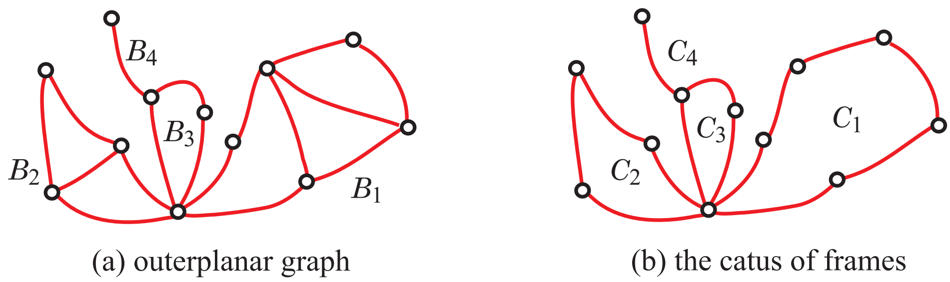


Figure 1: (a) An outerplanar graph; (b) The cactus obtained from the outerplanar graph in (a) by deleting the chords, where C_i denote the frame of each block B_i of the graph in (a).

Throughout the paper, we denote an instance of two-page book embedding problem by a graph $G = (V, E_1 \cup E_2)$ with a partition of E into E_1 and E_2 (i.e., $E_1 \cup E_2 = E$ and $E_1 \cap E_2 = \emptyset$), where two vertices may be joined by two edges $e \in E_1$ and $e' \in E_2$. We call the edges in E_1 (resp., E_2) *red edges* (resp., *blue edges*). A subgraph H in G is called *red* (resp., *blue*) if $E(H)$ consists of only red (resp., blue) edges. A vertex to which only red (resp., blue) edges are incident is called an *r-vertex* (resp., *b-vertex*). A vertex to which both red and blue edges are incident is called a *br-vertex*. A *2-page book embedding*

(2PB-embedding, for short) π of a graph $G = (V, E_1 \cup E_2)$ is a linear ordering of the vertices such that all vertices are placed in this order on a spine and all red edges are drawn above the spine and all blue edges are drawn below the spine without any edge-crossings. See Fig. 2(a) and (b) for examples of 2-page book embeddings of the same graph G.

In a 2PB-embedding π of a graph $G = (V, E_1 \cup E_2)$, if we join the first and last vertices on the spine with a new curve so that the spine together with the curve forms a simple closed curve which encloses all red edges but no blue edges (see Fig. 2(a)). Thus, a 2PB-embedding π can be regarded as a plane embedding Γ of G in which a simple closed curve λ visits each vertex exactly once without intersecting any edge and encloses all red edges but no blue edges. We call such a curve λ a *separating curve* of the embedding Γ. Note that the first and last vertices appear on the outer facial cycle in the plane embedding Γ. However by choosing a new outer face, any vertex v can appear along the outer facial cycle. This does not change the combinatorial embedding, and thereby the vertex v can appear as the first vertex on the spine in the 2PB-embedding obtained from the resulting plane embedding Γ'. See Fig. 2(b), where the vertices v_3 and v_2 are chosen as the first and last vertices on the spine.

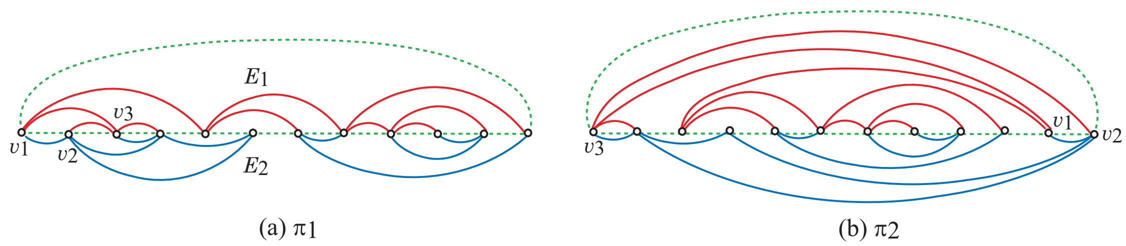


Figure 2: (a) A 2PB-embedding π_1 of a planar graph $G = (V, E_1 \cup E_2)$; (b) A 2PB-embedding π_2 of G obtained from π_1 by choosing the vertices v_3 and v_2 as the first and last vertices on the spine.

3 Canonical Instances

In this subsection, we give a linear-time algorithm for converting a given instance into another instance with a special structure, called a "canonical" instance, without changing the 2PB-embdedability.

Clearly an instance $G = (V, E_1 \cup E_2)$ admits a 2PB-embedding only when G is planar, and each E_i induces an outerplanar graph. If $E_1 = \emptyset$ or $E_2 = \emptyset$, then G has a 2PB-embedding if and only if G is outerplanar. In what follows, we assume that $E_1 \neq \emptyset \neq E_2$. Also we assume that G is not a simple cycle (otherwise it always admits a 2PB-embedding).

An instance $G = (V, E_1 \cup E_2)$ is *canonical* if

(i) G is a simple, biconnected and planar graph, but G is not a simple cycle;

(ii) Each E_i induces a cactus $(V, E_i \neq \emptyset)$; and

(iii) Each br-vertex of G is of degree 2.

We first show how G can be assumed to be biconnected.

Lemma 2 *A planar graph $G = (V, E_1 \cup E_2)$ admits a 2PB-embedding π with partition E_1 and E_2 if and only if each block H of G has a 2PB-embedding π_H for the partition $E(H) \cap E_1$ and $E(H) \cap E_2$ of $E(H)$.*

Proof. Since the only if part is trivial, we consider the if part. Let $H_1, H_2, \dots, H_p$ be the blocks of G indexed so that each H_i, $i = 2, 3, \dots, p$ has a common vertex v_{H_i} with some block H_j with $j < i$, where such v_{H_i} is unique to H_i and is called the parent of H_i. Assume that each block H_i of G has a 2PB-embedding π_{H_i} with the partition $E(H_i) \cap E_1$ and $E(H_i) \cap E_2$. As we have observed, we can regard each embedding π_{H_i} as a plane embedding of H_i, and by choosing the outer face so that the parent v_{H_i} appears on the boundary, we can obtain a 2PB-embedding π'_{H_i} of H_i in which v_{H_i} appears as the first vertex on the spine. Let $\pi := \pi'_{H_1}$. For each $i = 2, 3, \dots, p$, we place the 2PB-embedding π'_{H_i} in the space between vertex v_{H_i} and the vertex next to v_{H_i} in the current 2PB-embedding π, letting π'_{H_i} share the vertex v_{H_i} with π. In this way, we can construct a desired 2PB-embedding π of G. □

Now we can assume that an instance $G = (V, E_1 \cup E_2)$ is biconnected. We next show how each induced graph (V, E_i) from G is assumed to be a cactus. Let Γ_i be an outerplane embedding of (V, E_i) (where all vertices in V appear along the outer boundary). In Γ, each block of (V, E_i) is either a single edge or a cycle Q with some chords, where each chord is drawn within the cycle. By Lemma 1, in any such embedding of (V, E_i), chords are drawn inside the corresponding cycle and no other edges/vertices are included within the cycle. On the other hand, in any 2PB-embedding of G, such a cycle (frame) C with $E(Q) \subseteq E_i$ is drawn without surrounding any edge in E_j $(j \neq i)$ in its interior. This means that we can remove all the chords in the embedding Γ_i $(i = 1, 2)$ in the sense that we can put back them into any 2PB-embedding of the resulting instance without creating any edge crossings. Hence we can assume that each (V, E_i) is a simple cactus.

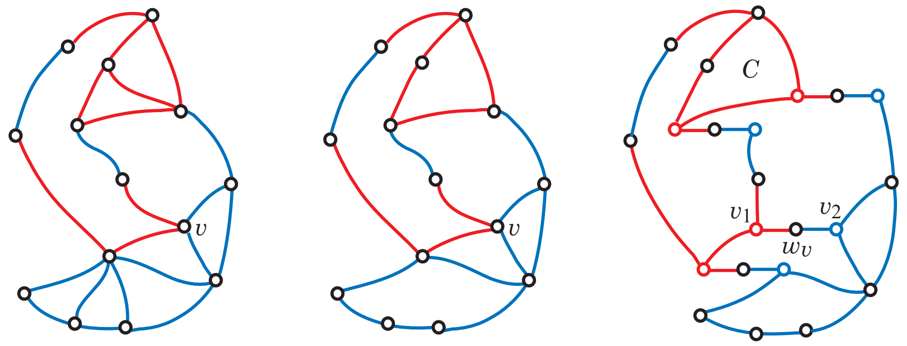
We transform a biconnected graph $G = (V, E = E_1 \cup E_2)$ into a canonical instance of 2PB-embedding problem as follows.

Definition 1.

Step 1. For each outerplanar graph (V, E_i), $i = 1, 2$, remove the chords of each block B to obtain a cactus (V, E'_i) in the resulting instance $G' = (V, E'_1 \cup E'_2)$ (see Fig. 3(a) and (b)).

Step 2. Let V_{br} be the set of br-vertices $v \in V$ of degree ≥ 3 in G'.
(i) Replace each $v \in V_{br}$ of degree ≥ 3 with three vertices v_1, w_v and v_2 joined by a new red edge (v_1, w_v) and a new blue edge (v_2, w_v); and
(ii) For each $i = 1, 2$, change the end vertices of each edge $(u, v) \in E'_i$ to (u_i, v_i) (see Fig. 3(b) and (c)). Let $G'' = (V'', E''_1 \cup E''_2)$ be the resulting graph.

Lemma 3 *Let G'' be the canonical instance obtained from a biconnected instance $G = (V, E_1 \cup E_2)$ by Definition 1. Then G admits a 2PB-embedding if and only if G'' admits a 2PB-embedding. Furthermore a 2PB-embedding of G'' can be converted into a 2PB-embedding of G in linear time.*



(a) G (b) G' (c) G''

Figure 3: (a) a biconnected instance $G = (V, E_1 \cup E_2)$; into a canonical instance $\widetilde{G}$ in (d) by ; (b) a biconnected instance $G' = (V, E'_1 \cup E'_2)$ with cactus (V, E'_i), $i = 1, 2$; and (c) a canonical instance $G'' = (V'', E''_1 \cup E''_2)$ with cactus (V, E''_i), $i = 1, 2$;

Proof. The correctness of Step 1 of Definition 1 has been discussed. We show the correctness of Step 2.

Only if part: Assume that G' admits a 2PB-embedding Γ. In Γ, we replace each vertex v with three vertices v_1, w_v and v_2 in this order on the spine, adding a red edge (v_1, w_v) on the red edge page and a blue edge (v_2, w_v) on the blue edge page, and changing the end vertices of each edge $(u, v) \in E_i$ to (u_i, v_i). By taking a sufficiently small space for the replacement for each vertex v, we can modify Γ into a 2PB-embedding of G''.

If part: Assume that G'' admits a 2PB-embedding Γ. See Fig. 4(a). We modify Γ into another a 2PB-embedding of G'' so that the three vertices v_1, w_v and v_2 for each $v \in V$ appear consecutively (possibly in a different order) in Γ. Then the resulting embedding can be converted into a 2PB-embedding of G' by contacting the three vertices v_1, w_v and v_2 for each $v \in V$ into a single vertex v. Suppose that there is some other vertex x ($\neq v_2$) between v_1 and w_v on the spine. Then we change the position of v_1 to a new position between the current positions of v_1 and w_v such that no other vertex x ($\neq v_2$) appears between v_1 and w_v. Since v_1 and w_v are joined by a red edge before the change, the red edge page has a space for the rest of red edges incident to v_1 to be drawn without creating any crossing with other edges. See Fig. 4(b). We can change the position of vertex v_2 analogously. By applying the method for all $v \in V$, we can obtain a desired embedding for G'. Note that the modification can be executed just by tracing the red edge (v_1, w_v) (resp., (v_2, w_v)) and the other red (resp., blue) edges incident to v_1 for each $v \in V$, and it can be implemented to run in linear time. □

A canonical instance G is simple, since G is not a cycle and it has no pair of blue and red multiple edges. Also, if there are a red u, v-path and a blue u, v-path for some vertices $u, v \in V$, then u and v are br-vertices of degree 2. In the following two sections, a given instance is assumed to be canonical.

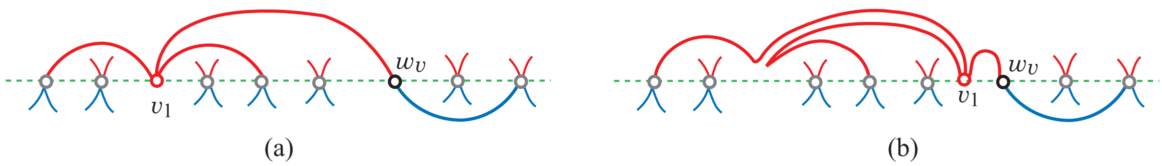


Figure 4: (a) A 2PB-embedding Γ of G''; (b) A modified 2PB-embedding so that v_1 is placed next to w_v.

4 Disjunctive and Splitter-free Plane Embeddings

In a plane embedding Γ of G, a red (resp., blue) cycle Q of G is called a *splitter* if each of the two regions obtained by the cycle Q contains a vertex $v \in V - V(Q)$ or a blue (resp., red) edge.

In Γ, a vertex v is called *disjunctive* if for each $i = 1, 2$, all the edges in $E_i(v; G)$ appear consecutively around v. We call Γ *disjunctive* if all vertices in V are disjunctive.

Theorem 4 *Let $G = (V, E_1 \cup E_2)$ be a biconnected planar graph with $E_1 \neq \emptyset \neq E_2$ (not necessarily canonical). Then G admits a 2PB-embedding π if and only if G admits a disjunctive and splitter-free plane embedding Γ. Moreover, a 2PB-embedding π of G can be obtained from a disjunctive and splitter-free plane embedding Γ of G in linear time.*

Proof. As observed above, a 2PB-embedding π of G can be regarded as a plane embedding which admits a separating curve λ, where we treat λ as an oriented curve so that the red edges appear on our left hand side when we traverse λ along its orientation. Conversely, if a plane embedding admits such a separating curve λ, then we can obtain a 2PB-embedding π of G.

Only if part: Assume that G has a 2PB-embedding π. Let λ be a separating curve of π. The curve λ visits each vertex exactly once and it encloses only red edges. Hence the plane embedding π is disjunctive at any vertex. If the plane embedding π has a red splitter, then λ cannot visit a vertex (or a blue edge) enclosed by the splitter and a vertex (or a blue edge) outside the splitter, because λ cannot intersect the splitter due to the disjunctive condition on the vertices in the splitter. Hence π has no red splitter. Similarly, π has no blue splitter.

If part: Assume that G has a disjunctive and splitter-free plane embedding Γ. It suffices to show that Γ admits a separating curve λ. For this, we construct an Eulerian plane digraph G^* and its Eulerian trail λ by the following procedure.

Since G is biconnected, the facial cycle of each face in Γ is a simple cycle.

Step 1. Let F_{br} be the set of faces of Γ whose facial cycles contain both red and blue edges. Place a new vertex v_f inside each face f in F_{br}.

Step 2. Let V_{br} be the set of br-vertices of G. For each br-vertex $v \in V_{\mathrm{br}}$, there exists a unique pair of faces f and f' sharing the vertex v such that all the red (resp., blue) edges incident to v appear in the clockwise order after visiting face f (resp., f') around v, because Γ is disjunctive. We join such two vertices v_f and $v_{f'}$ via a new directed edge (v_f, v) and $(v, v_{f'})$ (see Fig. 5(b)).

Step 3. We call all the new edges introduced in Steps 2 *green edges*. Let $G^* = (V_{\rm br} \cup V', E^g)$ be the Eulerian plane bipartite digraph that consists of all the green edges and their end vertices, where V' denotes the set of the end vertices v_f of all green edges; G^* is a plane, bipartite and Eulerian digraph which has no isolated vertices.

Step 4. Convert G^* into a simple directed cycle λ consisting of all the green edges without self-intersection in the plane, by splitting each vertex $v_f \in V'$ into $\deg(v_f; G^*)/2$ vertices of indegree 1 and outdegree 1. Such a cycle $\lambda_{\rm br}$ visits each vertex $v \in V_{\rm br}$ exactly once and encloses all the red edges but no blue edges (see Fig. 5(c)).

Step 5. Each b-vertex $u \in V - V_{\rm br}$ is on the facial cycle of a face $f_u \in F_{\rm br}$, because otherwise the union of the faces that contain v would include a blue splitter Q that separates v from a vertex or a red edge outside the region Q. Symmetrically, each r-vertex $u \in V - V_{\rm br}$ is on the facial cycle of a face $f_u \in F_{\rm br}$. We let the cycle $\lambda_{\rm br}$ visit each b-, or r-vertex $u \in V - V_{\rm br}$ while $\lambda_{\rm br}$ passes through the face $f_u \in F_{\rm br}$. The resulting cycle $\lambda_{\rm br}$ will be a separating cycle of Γ (see Fig. 5(d)).

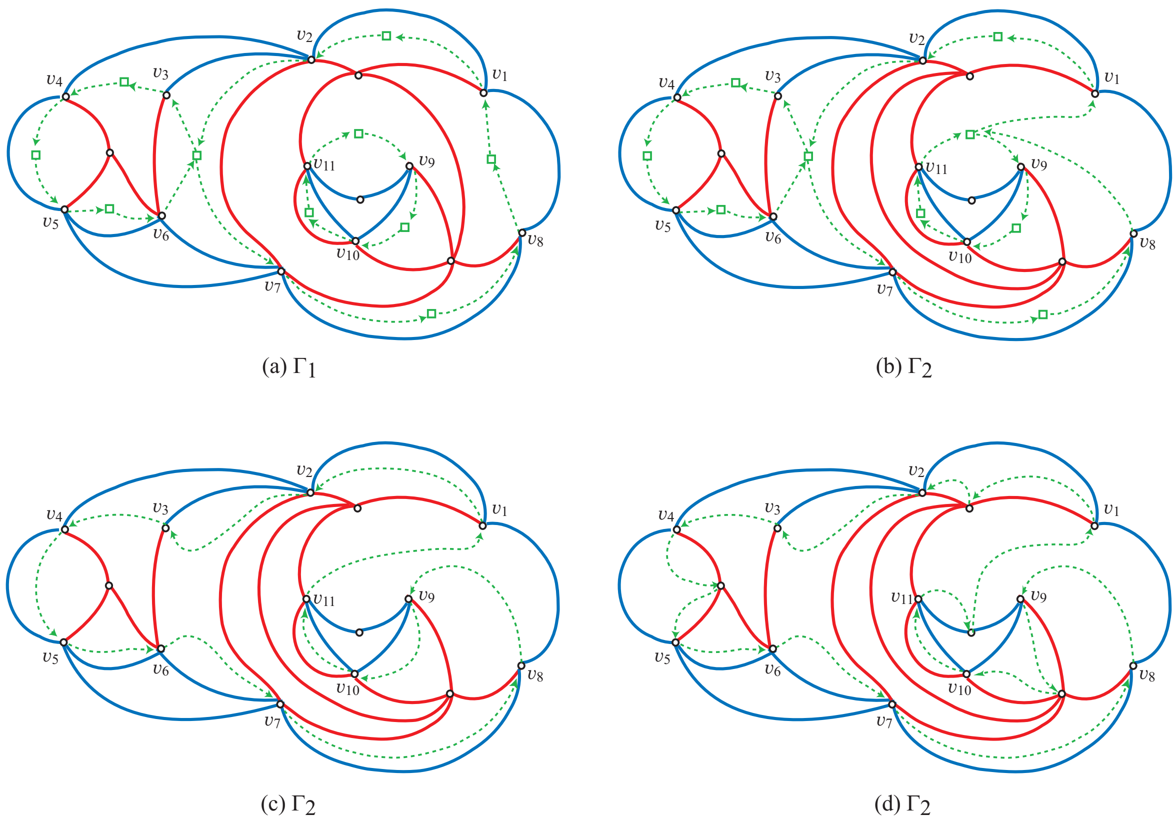
To complete the proof, it suffices to show that Step 4 can be executed. For this, we first show that the plane Eulerian digraph G^* is connected.

By the construction of the plane digraph G^*, for each vertex $v_f \in V'$, the incoming and outgoing green edges incident to v_f appear alternately around v_f. Hence we can traverse the boundary of G^* following the edge directions. Thus the boundary of each component of G^* is a directed cycle.

To derive a contradiction, assume that G^* has two components H and H' (as shown in Fig. 5(a)), and let B and B' denote the boundary of H and H', respectively, where H' is not contained in the interior of B without loss of generality. Let the direction along B be the clockwise order (the other case can be treated symmetrically). By the construction of G^*, only red edges are adjacent to B outside B (as shown in Fig. 5(a)), and each vertex v_f in B has no blue edges on the corresponding facial cycle $f \in F_{\rm br}$. This implies that the union of these facial cycles contain a red cycle Q of G that separates a blue edge in $E(H)$ and a vertex $v' \in V(H')$. Hence Q is a splitter of Γ, contradicting that G is splitter-free. Therefore G^* is connected (see Fig. 5(b)). Since G^* is a connected Eulerian digraph, it has an Eulerian trail (a simple directed cycle consisting of all the green edges). We show that there is an Eulerian trail which has no self-intersection in the plane.

Based on this observation, we partition the set E^g of green edges into $E^g_1, E^g_2, \ldots, E^g_p$ as follows. Let E^g_1 be the set of the edges in the boundary, and $G^*_1 = G^* - E^g_1$. Analogously we can traverse the boundary of each component of G^*_1 following the edge directions. We repeatedly define E^g_i as the set of edges in the boundaries of the components of $G^* - (E^g_1 \cup \cdots \cup E^g_{i-1})$ until $G^* - (E^g_1 \cup \cdots \cup E^g_i)$ has no edge for some $i = p$. Each set E^g_i gives a collection of Eulerian trail with no self-intersection. The set of all these trails forms a rooted tree structure in which the trail in E^g_1 is the root and a trail in E^g_i is a child of a trail E^g_{i-1} if these trails share a vertex v_f. We can combine all these trails into a single Eulerian trail $\lambda_{\rm br}$ with no self-intersection by traversing the trails in a DFS manner in the tree structure. The above procedure can be implemented to run in linear time by maintaining trails as doubly-linked lists so that two trails with a common vertex can be merged in constant time. This proves the if part.

It is not difficult to see that Steps 1-5 can be implemented to run in linear time to



(a) Γ_1 (b) Γ_2

(c) Γ_2 (d) Γ_2

Figure 5: Two embeddings of a planar graph $G = (V, E_1 \cup E_2)$; (a) a plane embedding Γ_1 of G which has no separating curve λ; (b) A disjunctive and splitter-free plane embedding Γ_2 of G, where the circles represent the vertices in G, the squares represent the new vertices v_f for the faces $f \in F_{\rm br}$, and the dashed arrows represent the green edges defined in the proof of Theorem 4; (c) A closed curve $\lambda_{\rm br}$ visiting all br-vertices of G; (d) A closed curve λ visiting all vertices of G.

obtain a separating curve λ, from which a 2PB-embedding of G can be obtained in linear time. This proves the second statement in the theorem. $\square$

Notice that any plane embedding of a canonical instance is disjunctive, since red and blue edges meet at a br-vertex of degree 2.

5 Reduction to Planarity Test

Let $G = (V, E_1 \cup E_2)$ be a canonical instance. In a plane embedding Γ of a canonical instance G, a red cycle Q of G is a splitter if and only if Q is not a facial cycle of Γ. The only-if part is immediate, since no facial cycle can separate two vertices/edges. The if part follows from that each of the two regions R_1 and R_2 by a non-facial red cycle Q must contain a blue edge, because otherwise R_i contains a red path joining two vertices on Q, contradicting that (V, E_1) is a cactus. Symmetrically a blue cycle becomes a splitter in Γ whenever it is not a facial cycle of Γ. We call a plane embedding Γ of G *proper* if any red/blue cycle of G appears as a facial cycle of Γ.

Detecting a proper embedding Γ of G can reduced to the standard planarity testing of an augmented graph as follows. For each cycle C in the cactus (V, E_i), $i = 1, 2$, we subdivide each edge e in C with a new vertex w_e, create a new vertex v_C, and add new edges (v_C, w_e), $e \in E(C)$. Fig. 6(a) shows the graph $\widetilde{G}$ obtained from the canonical instance in Fig. 3(c). Let $\widetilde{G} = (\widetilde{V}, \widetilde{E}_1 \cup \widetilde{E}_2)$ denote the resulting graph, where $\widetilde{E}_i$ is the set of edges obtained by subdividing an edge in E_i or introduced to augment a cycle in the cactus (V, E_i).

Lemma 5 *A canonical instance $G = (V, E_1 \cup E_2)$ admits a proper embedding if and only if $\widetilde{G}$ is planar. A proper embedding of G can be obtained from a plane embedding of $\widetilde{G}$ in linear time.*

Proof. Only-if part: Let Γ_G be a proper embedding of G. Then each cycle block C of the cactus (V, E_i) surrounds no edge in E_j $(j \neq i)$. Hence, we can draw the newly added vertices v_C, and w_e, $e \in E(C)$ and edges between them inside the empty region of C without creating edge crossings. Thus the resulting embedding is a plane embedding of $\widetilde{G}$.

If part: Since G is biconnected, $\widetilde{G}$ is also biconnected. Let $\Gamma_{\widetilde{G}}$ be a plane embedding of $\widetilde{G}$, where without loss of generality that the outer facial cycle f^o contains a br-vertex z of G. Note that z is not in any cycle of a cactus (V, E_i). Let $\Gamma_{\widetilde{E}_i}$ denote the plane embedding induced from $\Gamma_{\widetilde{G}}$ by the edges in $\Gamma_{\widetilde{E}_i}$. We first show that for each block C in (V, E) augmented with vertex v_C, the vertex v_C is surrounded by the cycle C (i.e., C separates v_C from f^o) in $\Gamma_{\widetilde{E}_i}$. If for some C, both v_C and f^o are outside C, then by the way of augmentation, only one vertex $u \in V(C)$ can be adjacent to vertices outside C (see Fig. 6(b)), and u would be a cut-vertex separating v_C and z in $\widetilde{G}$, contradicting the biconnectivity of $\widetilde{G}$. Now v_C of each cycle is located inside the subdivided cycle C in $\Gamma_{\widetilde{E}_i}$. Next we show that no other vertex than v_C is located inside the subdivided cycle C in $\Gamma_{\widetilde{G}}$. Assume that for some C, a vertex y $(\neq v_C)$ is inside the subdivided C in $\Gamma_{\widetilde{G}}$ (see Fig. 6(c)). Since G is connected, y is connected to a vertex u in the the subdivided C. Let u be the one closest to y among such u. Then in G, $u \in V(C)$, and u is adjacent to two edges $e, e' \in E(C)$. Now in the plane embedding $\Gamma_{\widetilde{G}}$, the cycle $(u, w_e, v_C, w_{e'})$ surround the set Y of all vertices reachable from y without passing through u. This, however, implies that u is a cut-vertex separating y and z in $\widetilde{G}$, a contradiction. Hence, each subdivided C encloses no other vertex than v_C in $\Gamma_{\widetilde{G}}$.

Let Γ be the plane embedding of G induced from $\Gamma_{\widetilde{G}}$ by G; i.e., remove the augmented vertices v_C and ignore the introduced vertices w_e for all cycles C in cacti (V, E_1) and (V, E_2). Clearly, each C of a cactus (V, E_i) encloses no edges/vertices in Γ. Since each red (resp., blue) cycle Q in G is a (facial) cycle of (V, E_1) (resp., (V, E_2)), the resulting embedding Γ is proper. □

Since testing planarity and constructing a plane embedding if any can be done in linear time, we can find a 2PB-embedding of a given instance if any in linear time. When $\Gamma_{\widetilde{G}}$ is not planar, it contains a subdivision of K_5 or $K_{3,3}$. However, such a subgraph is not a direct evidence of a given infeasible instance due to the augmentation. In the next section, we give an algorithm that detects a forbidden subgraph of a given instance.

6 Forbidden Subgraphs to Two-page Book Embeddings

In this paper, we also characterize the instances G that do not admit 2PB-embeddings.

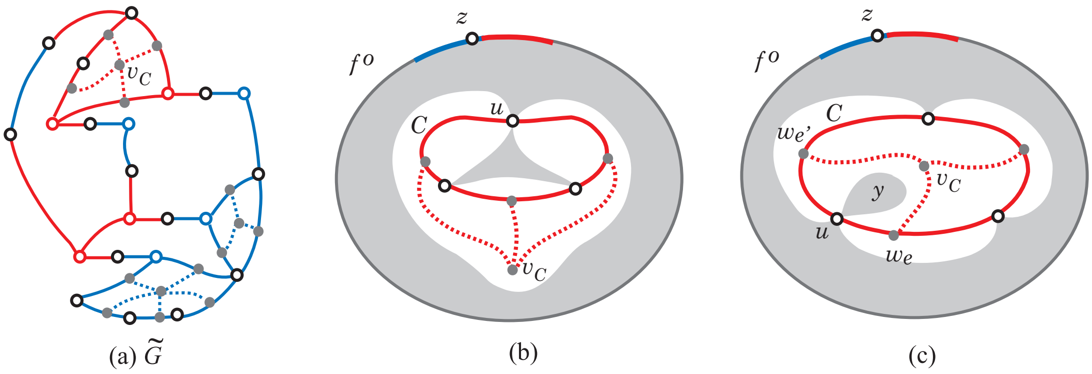


Figure 6: Illustration for a plane embedding of the augmented graph $\widetilde{G}$; (a) Graph $\widetilde{G}$ obtained from the graph in Fig. 3(c) by augmenting each cycle in red and blue cacti with stars; (b) f^o and v_C are outside the subdivided cycle C; (c) other vertex y than v_C is inside the subdivided cycle C.

A graph H is called *pseudo-triconnected* if a subdivision of a triconnected graph G. Observe that a graph H is pseudo-triconnected if and only if H has at least three vertices of degree ≥ 3 and every two vertices u and v of degree ≥ 3 in H has three internally disjoint paths. We call a pseudo-triconnected subgraph S of G a *forbidden subgraph* if a cycle Q with E_i in a plane embedding of S separates two edges $e_1, e_2 \in E_i$ ($i \in \{1,2\} - \{j\}$). By the uniqueness of plane embedding of S, such a subgraph S (and hence G) cannot admit a 2PB-embedding. In this paper, we show that the converse is true establishing the following result.

Theorem 6 *Let $G = (V, E_1 \cup E_2)$ be a planar graph with a partition E_1 and E_2 of $E(G)$ such that each E_i induces a simple and outerplanar subgraph (V, E_i). Then G admits a 2PB-embedding if and only if there is no forbidden subgraph in G. Furthermore, either a 2PB-embedding or a forbidden subgraph of G can be found in linear time.*

A completer characterization of forbidden subgraphs is useful since it can answer the solution to restricted graph classes. For example, when each of E_1 and E_2 induces a forest, G cannot have a forbidden subgraph. Hence the theorem implies the next.

Corollary 7 *Let $G = (V, E_1 \cup E_2)$ be a graph with a partition E_1 and E_2 of $E(G)$ such that each E_i induces a forest (V, E_i). Then G admits a 2PB-embedding if and only if G is planar. Furthermore, a 2PB-embedding of G if any can be found in linear time.*

To prove Theorem 6, we first design a linear-time algorithm for detecting a forbidden subgraph of a given canonical instance. It is not difficult see that even if a canonical instance G is obtained from the original instance G^* by Definition 1, a forbidden subgraph S of G gives a forbidden subgraph of G^*. We then prove that a canonical instance with no forbidden subgraphs admits a proper embedding (and hence a 2PB-embedding) by designing a linear-time algorithm for constructing a proper embedding for such an instance.

6.1 SPQR tree representations

To consider the all possible plane combinatorial embeddings of a biconnected graph, we use the SPQR tree by di Battista and Tamassia [9, 10] (see Appendix 1 for detail). The SPQR tree $\mathcal{T}$ of a biconnected graph G represents the adjacency of the triconnected components of G. Each node ν in $\mathcal{T}$ corresponds to a triconnected component of G, and is associated with a graph $\sigma(\nu)$ called the *skeleton* of ν, such that a subdivision of appears as a subgraph of G (hence $V(\sigma(\nu)) \subseteq V(G)$). For each edge $e = (u, v) \in E_\nu$, G has an induced subgraph G_e which shares $\{u, v\}$ as a cut-pair with the complementary part $G - (V(G_e) - \{u, v\})$. Hence the skeleton $\sigma(\nu)$ provides an abstract structure of the entire graph G; there is a subgraph S of G that is a subdivision of $\sigma(\nu)$, which is obtained by replacing each virtual edge (u, v) with a u, v-path of G. For each R-node ν, its skeleton $\sigma(\nu)$ is triconnected and its plane embedding is unique up to choices of outer faces.

For each edge of the skeleton of a node ν of the SPQR tree$\mathcal{T}$ of a canonical instance $G = (V, E_1 \cup E_2)$, the graph G_e is called an *r-graph* (resp., *b-graph*) if it has a red (resp., blue) u, v-path. Note that G_e cannot have both a red u, v-path and a blue u, v-path, since u and v are br-vertices of degree 2 and cannot be adjacent to other vertices in $V - V(G_e)$. A virtual edge e is called an *r-edge* (resp., *b-edge*) if G_e is an *r-graph* (resp., *b-graph*). Again no virtual edge can be an r-edge and b-edge at the same time. We also treat a red (resp., blue) real edge as an r-edge (resp., b-edge). Fig. 7(b) shows the skeleton $\sigma(\nu)$ of the R-node ν with $V(\sigma(\nu)) = \{v_1, v_3, v_5, v_7, z_1, z_2, z_3\}$ of the graph in Fig. 7(a). For a subgraph H of the skeleton $\sigma(\nu)$ of a node ν, let $E^r(H)$ (resp., $E^b(H)$) denote the set of r-edges (resp., b-edges) in H. Note that each of $E^r(H)$ and $E^b(H)$ induces a cactus, since each (V, E_i) is a cactus. Hence each P-node ν of $\mathcal{T}$ satisfies

$$|E^r(\sigma(\nu))| + |E^b(\sigma(\nu))| \leq 2. \tag{1}$$

6.2 Splitters and forbidden subgraphs

Now we examine the structure of red/blue splitters in canonical instances. A cycle Q' in the skeleton $\sigma(\nu)$ of a node ν is called an *r-cycle* (resp., *b-cycle*) if $E(Q') \subseteq E^r(\sigma(\nu))$ (resp., $E(Q') \subseteq E^b(\sigma(\nu))$).

For example, $Q' = (v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8)$ in Fig. 7(b) is a non-facial r-cycle in the skeleton $\sigma(\nu)$ of the R-node ν.

In fact, splitters and forbidden subgraphs are equivalent in the following sense.

Lemma 8 *Let $G = (V, E_1 \cup E_2)$ be a canonical instance.*
(i) *Let Q' be a non-facial r-cycle in the skeleton $\sigma(\nu)$ of an R-node ν. Then the subgraph S of G obtained from $\sigma(\nu)$ by replacing each virtual edge $e = (u, v) \in E(Q')$ (resp., $e = (u, v) \in E(\sigma(\nu)) - (Q')$) with a u, v-path of Q (resp., G) is a forbidden subgraph of G.*
(ii) *Let S be a forbidden subgraph such that a red cycle Q in S separates two blue edges e_1, e_2 of G. Then for the R-node ν such that $V(\nu)$ contains all the veritces of degree ≥ 3 of S, the set of r-edges $e \in E^b(\sigma(\nu))$ with $E(G_e) \cap Q \neq \emptyset$ is a non-facial r-cycle Q' in the skeleton $\sigma(\nu)$.*

Proof. (i) It suffices to show that the r-cycle Q' in $\sigma(\nu)$ corresponding to Q separates two blue edges of G. Since Q' is a non-facial r-cycle in $\sigma(\nu)$, Q' separates two non-r-edges e_1, e_2

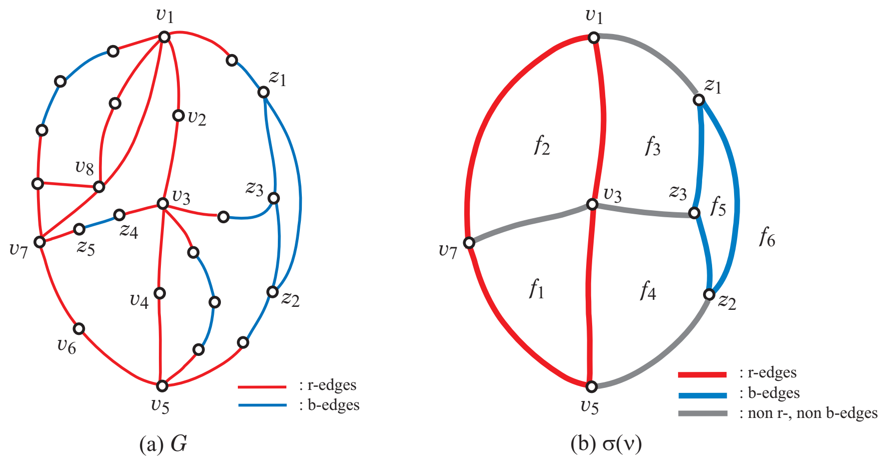


Figure 7: (a) A graph G which has a non-facial r-cycle $Q' = (v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8)$; (b) The skeleton $\sigma(\nu)$ of the R-node ν with $V(\sigma(\nu)) = \{v_1, v_3, v_5, v_7, z_1, z_2, z_3\}$ of G in (a).

in the two regions R_1 and R_2 in a plane embedding γ_ν of $\sigma(\nu)$. This means that R_i contains at least one blue edge of G.

(ii) Since S is a pseudo-triconnected subgraph of G, there is an R-node ν in which all vertices of degree ≥ 3 of S appear. Then $\sigma(\nu)$ has the r-cycle Q' corresponding to Q. By the uniqueness of embedding of S and $\sigma(\nu)$, Q' still separates the blue edges e_1 and e_2, and this means that Q' is not a facial cycle in a plane embedding γ_ν of $\sigma(\nu)$. This shows that Q' is a non-facial r-cycle in the skeleton $\sigma(\nu)$ of the R-node ν. $\square$

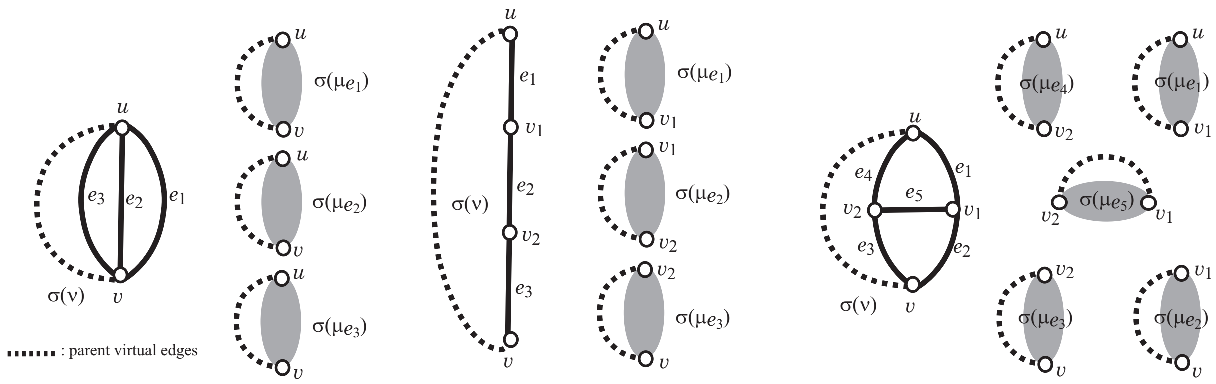
Lemma 9 *Given $E^r(\sigma(\nu))$ (resp., $E^b(\sigma(\nu))$) for the skeleton $\sigma(\nu)$ of an R-node ν, testing if there is a non-facial r-cycle (b-cycle) in a plane embedding γ_ν of $\sigma(\nu)$ can be done in $O(|E(\sigma(\nu))|)$ time.*

Proof. We consider the case of r-cycles (the case of b-cycles can be treated symmetrically). Since the red graph (V, E_1) is a cactus and no two vertices in V are joined by three vertex-disjoint paths. This means that the r-edges in the skeleton $\sigma(\nu)$ induce a cactus $(V(\sigma(\nu)), E^r(\sigma(\nu)))$. Since no two cycles share an edge in a cactus, the total size of all r-cycles in the cactus $(V(\sigma(\nu)), E^r(\sigma(\nu)))$ is $O(|E(\sigma(\nu))|)$. Hence even if we test whether each r-cycle in $\sigma(\nu)$ is a facial cycle of γ_ν or not, the total time complexity is $O(|E(\sigma(\nu))|)$. $\square$

Lemma 9 implies that once we know $E^r(\sigma(\nu))$ (resp., $E^b(\sigma(\nu))$) for all nodes ν in the SPQR tree, a non-facial r-/b-cycle in the skeleton of an R-node can be found in linear time. In the rest of the section, we show how to compute the r-edges and b-edges in the skeletons of all nodes in the SPQR tree.

6.3 Rooted SPQR trees

When a node in the SPQR tree of G is designated as the root, the SPQR tree is treated as a rooted tree, which defines a parent-child relationship among the nodes of the tree. For a node ν, a node μ adjacent to ν in SPQR tree is called the *parent* of ν if μ is closer to the root than ν is, and is called a *child* of ν otherwise. Let $Ch(\nu)$ denote the set of all children of ν. We denote the graph formed from $\sigma(\nu)$ by deleting its *parent virtual edge* $\mathrm{pe}(\nu)$ as $\sigma^-(\nu)$, if ν is not the root of $\mathcal{T}$. Let $G^-(\nu)$ denote the subgraph induced from G by the set of all vertices in the graphs $\sigma^-(\mu)$ for all descendants μ of ν, including ν itself. For each virtual edge $e \in E(\sigma(\nu))$, let $\mu_e \in Ch(\nu)$ denote the corresponding child node. For each non-root node ν, we denote by $\overline{G}(\nu)$ the graph $G - (V(G^-(\mu)) - \{u, v\})$. See Fig. 8 for illustrations of the skeleton of a node *nu* and its children.



(a) The skeleton σ(ν) of a P-node ν and the skeletons of its children μ_{e_i}

(b) The skeleton σ(ν) of an S-node ν and the skeletons of its children μ_{e_i}

(c) The skeleton σ(ν) of an R-node ν and the skeletons of its children μ_{e_i}

Figure 8: The skeleton $\sigma(\nu)$ of a ν and the skeletons $\sigma(\mu_{e_i})$ of its children $\mu_{e_i} \in Ch(\nu)$, where ν is a P-node in (a), an S-node in (b) and an R-node (c).

6.4 Computing the color of edges in the skeletons of all nodes along rooted SPQR trees

We first give an overview of the algorithm for computing the r-edges in the skeletons of all nodes in the SPQR tree of G (computing b-edges can be done symmetrically):
1. First choose a node as the root of the SPQR tree of G, which introduces a parent-child relationship among the nodes of the tree.
2. By traversing the rooted SPQR tree in a bottom-up manner, compute the r-edges in the skeleton $\sigma^-(\nu)$ of each node ν (except for the parent virtual edge $\mathrm{pe}(\nu)$), based on the computation of r-edges in the skeletons of children of ν.
3. By traversing the rooted SPQR tree in a top-down manner, identify the children $\mu \in Ch(\nu)$ of each node ν such that the parent virtual edge $\mathrm{pe}(\mu)$ is an r-edge.

Since G is not a simple cycle, the SPQR tree of G has an R- or P-node. We choose an R- or P-node ν^* as the root of the SPQR tree $\mathcal{T}$. For a leaf node ν in the rooted SPQR tree $\mathcal{T}$, we know that $E^r(\sigma^-(\nu))$ is the set of the red edges in the subgraph $G^-(\nu)$. The next lemma says that the r-edges in the skeletons $\sigma^-(\nu)$ of all other nodes in the SPQR tree $\mathcal{T}$ can be computed in linear time.

Lemma 10 *Given $E^r(\sigma^-(\mu))$ for a node ν in $\mathcal{T}$, the set $E^r(\sigma^-(\nu))$ of all r-edges in the skeleton $\sigma^-(\nu)$ except* $\mathrm{pe}(\nu)$ *can be computed in $O(|E(\sigma(\nu))| + \sum_{\mu\in Ch(\nu)} |E(\sigma(\mu))|)$ time.*

Proof. An edge $e = (u, v)$ in $\sigma^-(\nu)$ is an r-edge if and only if $G^-(\mu_e)$ of the corresponding child $\mu_e \in Ch(\nu)$ has a red u, v-path, which is equivalent to that skeleton $\sigma^-(\mu_e)$ has a u, v-path P_{uv} that consists of r-edges. Finding such a path P_{uv} in $\sigma^-(\mu_e)$ takes $O(|E(\sigma(\mu_e))|)$ time. A real edge e in $\sigma^-(\nu)$ is an r-edge if and only if it is a red edge. Therefore, the set $E^r(\sigma^-(\nu))$ of all r-edges can be obtained in the claimed time bound. □

For the root $\nu*$, we know $E^r(\sigma(\nu^*)$. The next lemma says that the parent virtual r-edges of the skeletons $\sigma(\nu)$ of all other nodes in the SPQR tree $\mathcal{T}$ can be computed in linear time.

Lemma 11 *Given $E^r(\sigma^-(\nu))$ for a node ν in $\mathcal{T}$, the set of all children $\mu \in Ch(\nu)$ such that* $\mathrm{pe}(\mu)$ *is an r-edge can be computed in $O(|E(\sigma(\nu))|)$ time.*

Proof. Observe that for each edge $e = (u, v) \in E(\sigma^-(\nu))$, its parent virtual edge $\mathrm{pe}(\mu_e)$ is an r-edge if and only if $\sigma(\nu) - \{e\}$ contains a path that consists of r-edges. We compute the connected components $C_1, \ldots, C_k$ of the graph $(V(\sigma(\nu)), E^r(\sigma(\nu)))$ induced from $\sigma(\nu)$ by the r-edges, and identify all bridges in the components C_i, $i = 1, \ldots, k$ (where C_i may consists of a single vertex). This can be done in linear time by the depth-first search algorithm. Then for each non-r-edge $e = (u, v) \in E(\sigma^-(\nu))$, $\mathrm{pe}(\mu_e)$ is an r-edge if and only if u and v belong to the same component C_i. On the other hand, for each r-edge $e = (u, v) \in E(\sigma^-(\nu))$, $\mathrm{pe}(\mu_e)$ is an r-edge if and only if e is not a bridge of the same component C_i to which u and v belong. Based on these characterizations, we can find all children $\mu \in Ch(\nu)$ such that $\mathrm{pe}(\mu)$ is an r-edge in $O(|E(\sigma(\nu))|)$ time. □

By Lemma 8, any forbidden subgraph can be found as a non-facial r- or b-cycle in the skeleton of an R-node. After we compute the r-edges in the skeletons of all nodes in the SPQR tree in linear time by Lemmas 10 and 11, we test if each R-node has a non-facial r-cycle in its skeleton in time linear of the size of the skeleton by Lemma 9, which takes $O(|V|+|E|)$ time in total over all R-nodes. Symmetrically we can find a non-facial b-cycle in the skeleton of an R-node in linear time. Hence finding a forbidden subgraph of a canonical instance, if any can be done in linear time.

To prove the theorem 6, the remaining task is to design a linear-time algorithm for constructing a proper embedding for a canonical instance with no forbidden subgraphs.

7 Constructing Proper Embeddings

In this section, we assume that a given canonical instance $G = (V, E_1 \cup E_2)$ has no forbidden subgraph, i.e., no non-facial r- or b-cycle in the skeleton of any R-node, and present a procedure for constructing a proper plane embedding.

In what follows, for a graph $H = \sigma^-(\nu)$ or $H = G^-(\nu)$ of each non-root node ν in $\mathcal{T}$, an embedding ψ of H means a plane embedding of the graph such that the both end vertices u and v of the parent virtual edge $\mathrm{pe}(\nu) = (u, v)$ of ν appear in the boundary of the plane embedding. When we traverse the boundary of ψ, we denote the path along the boundary from u to v (resp., from v to u) by $B_{u,v}(\psi)$ (resp., $B_{v,u}(\psi)$). The path $B_{u,v}(\psi)$ is called *r-rimmed* $B_{u,v}(\psi)$ is the unique r-u, v-path. We define *b-rimmed* boundaries symmetrically. Fig. 9(a) and (b) shows an embedding γ_ν of the skeleton $\sigma(\nu)$ and an embedding Γ_ν of the graph $G^-(\nu)$ for an R-node ν, where $B_{u,v}(\gamma_\nu)$ and $B_{u,v}(\Gamma_\nu)$ are r-rimmed while $B_{v,u}(\gamma_\nu)$ and $B_{v,u}(\Gamma_\nu)$ are both no r-rimmed.

An embedding ψ of H is called *proper* if (i) every r-cycle/b-cycle in ψ is a facial cycle; and (ii) $B_{u,v}(\psi)$ or $B_{v,u}(\psi)$ is r-rimmed (resp., b-rimmed) when $\mathrm{pe}(\nu)$ is an r-edge (resp., b-edge). A plane embedding ψ of $H = G$ or $H = \sigma(\nu)$ for the root node ν is called *proper* if every r-cycle/b-cycle in ψ is a facial cycle.

Assuming that each edge $e \in E(\sigma^-(\nu))$ of a node ν admits a proper embedding Γ_e of G_e, we show that a proper embedding Γ_ν of $G^-(\nu)$ can be obtained from these embeddings Γ_e. For this, we first observe that the skeleton of each node admits a proper embedding.

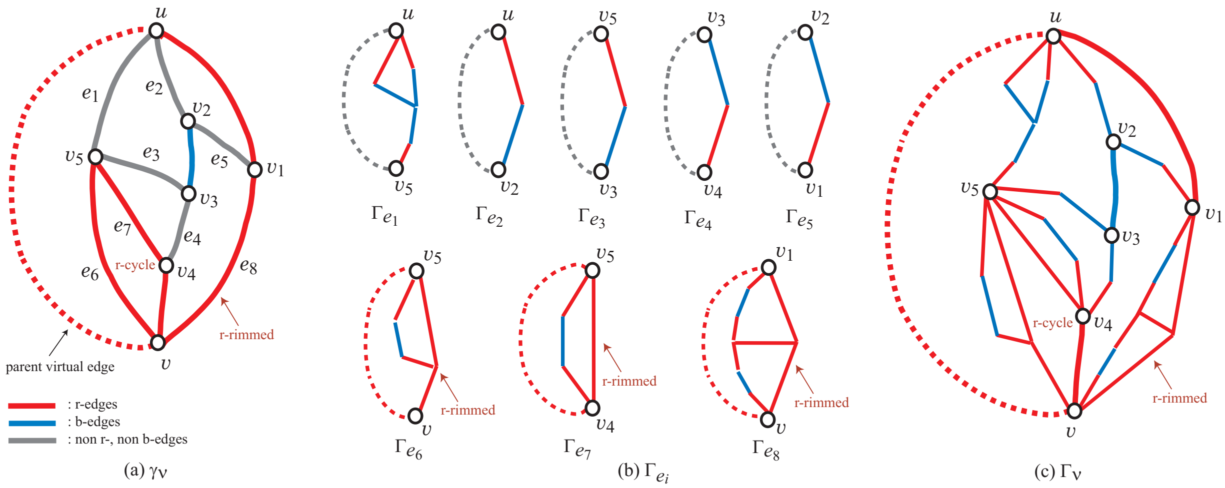


Figure 9: (a) A proper embedding γ_ν of the skeleton $\sigma(\nu)$ of an R-node ν, where $e_1, \ldots, e_8$ are the virtual edges of $\sigma(\nu)$; (b) A proper embedding Γ_{e_i} of subgraph G_e of each virtual edge e_i, $i = 1, 2, \ldots, 8$; and (c) A proper embedding Γ_ν of subgraph $G^-(\nu)$ obtained from γ_ν by replacing each virtual edge e_i with Γ_{e_i}.

Lemma 12 *Let G be a canonical graph with no forbidden subgraph, and ν be a node in the SPQR tree. Then:*

(i) *When ν is an S- or R-node or a P-node with $|E^r(\sigma(\nu))| + |E^b(\sigma(\nu))| \leq 1$, any plane embedding γ_ν of the skeleton $\sigma(\nu)$ of the root ν or of $\sigma^-(\nu)$ of a non-root ν is proper.*

(ii) *When ν is a P-node with $|E^r(\sigma(\nu))| = 2$ (resp., $|E^b(\sigma(\nu))| = 2$), a plane embedding γ_ν of the skeleton $\sigma(\nu)$ of the root ν or of $\sigma^-(\nu)$ of a non-root ν is proper if one of the two edges in $E^r(\sigma(\nu))$ (resp., $E^b(\sigma(\nu))$) appear consecutively (possibly one appears as $B_{u,v}(\gamma_\nu)$ and the other as the parent virtual edge $\mathrm{pe}(\nu)$).*

Proof. (i) The lemma is immediate for a P-node ν with $|E^r(\sigma(\nu))| + |E^b(\sigma(\nu))| \leq 1$ in (1) or an S-node.

Let ν be a non-root R-node (the case of root R-node can be treated analogously). Let γ_ν be a plane embedding of $\sigma^-(\nu)$ Since G has no forbidden subgraph, any r-/b-cycle of $\sigma(\nu)$ is a facial cycle of any plane embedding of $\sigma(\nu)$. Assume that the parent virtual edge $\mathrm{pe}(\nu)$ of ν is an r-edge (the case where the parent virtual edge of ν is a b-edge can be shown symmetrically). Hence the parent virtual edge $\mathrm{pe}(\nu)) = (u, v)$ is an r-edge in $\sigma(\nu)$. If $\sigma^-(\nu)$ has an r-u, v-path P, then $\mathrm{pe}(\nu)$ and P form an r-cycle Q', which again must be a facial cycle in a plane embedding of $\sigma(\nu)$; i.e., P is $B_{u,v}(\gamma_\nu)$ or $B_{v,u}(\gamma_\nu)$, and $\sigma^-(\nu)$ has no r-u,v-path other than P. This proves that γ_ν is proper.

(ii) For a P-node ν with $|E^r(\sigma(\nu))| = 2$ or $|E^b(\sigma(\nu))| = 2$, the embedding stated in the lemma is proper, since $\sigma(\nu)$ has exactly one r-cycle or b-cycle. □

Based on proper embeddings of all nodes in the SPQR tree, we show how to construct a proper embedding for $H = G$ traversing the rooted SPQR tree $\mathcal{T}$ in a bottom-up manner. We first construct proper embeddings for leaf nodes of $\mathcal{T}$. Then we construct a proper embedding of a non-leaf node ν by assembling the proper embeddings of all children of ν.

Leaf nodes: For each leaf S- or R-node ν of $\mathcal{T}$, any plane embedding of $G^-(\nu) = \sigma^-(\nu)$ is proper by Lemma 12. Note that there is no leaf P-node since G is simple.

Internal nodes: Let ν be an internal node of $\mathcal{T}$, $\mathrm{pe}(\nu) = (u, v)$ be the parent virtual edge of ν, and γ_ν be a proper embedding of $\sigma^-(\nu)$ in Lemma 12. For each virtual edge $e \in E(\sigma^-(\nu))$, let Γ_e denote a proper embedding of G_e.

(1) S-node: Let ν be an S-node. In this case, γ_ν is a single path joining u and v. Let Γ_ν be an embedding of $G^-(\nu)$ obtained from γ_ν by replacing each virtual edge $e \in E(\sigma^-(\nu))$ with Γ_e. If $G^-(\nu)$ is not an r-graph or a b-graph, then the resulting embedding Γ_ν of $G^-(\nu)$ is already proper. Assume that $G^-(\nu)$ is an r-graph (the case of a b-graph can be treated symmetrically). Then, (α) for each edge e in $B_{u,v}(\xi_\nu)$, we flip each Γ_e if necessary so that the r-rimmed boundary of ψ_e appears along the outer face of γ_ν. The resulting embedding Γ_ν of $G^-(\nu)$ is now proper.

(2) R-node: Let ν be an R-node, and γ_ν be a proper embedding of $\sigma^-(\nu)$, where we assume without loss of generality that $B_{u,v}(\gamma_\nu)$ is the unique r-u,v-path of $\sigma^-(\nu)$. See Fig. 9, where (a) shows a proper embedding γ_ν of $\sigma^-(\nu)$ of the R-node ν, and (b) shows a proper embedding Γ_{e_i} of G_{e_i} of each virtual edge $e_i \in E(\sigma(\nu))$. Consider the case where $G^-(\nu)$ is an r-graph and the parent virtual edge of ν is an r-edge (the other case can be treated analogously). We replace each virtual edge e in $\sigma^-(\nu)$ so that (α) for each edge e in $B_{u,v}(\gamma_\nu)$, the r-rimmed boundary of Γ_e of appears along the outer face of γ_ν; and (β) for each edge e in a facial r-cycle Q' of γ_ν, the r-rimmed boundary of Γ_e of appears facing the interior of Q'. Let Γ_ν be the resulting embedding of $G^-(\nu)$. See Fig. 9(c) for the resulting embedding Γ_ν obtained from the embeddings in Fig. 9(a) and (b). We see that Γ_ν is proper, since γ_ν and all ψ_e are proper.

(3) P-node: Let ν be a P-node. Let γ_ν be a proper embedding of $\sigma^-(\nu)$. That is, two edges in $E^r(\sigma(\nu))$ (resp., $E^r(\sigma(\nu))$), if any, appear consecutively (possibly one as $B_{u,v}(\gamma_\nu)$ and the other as the parent edge $\mathrm{pe}(\nu)$). We replace each virtual edge e in $\sigma^-(\nu)$ by Γ_e according to the same rules (α) and (β). We easily see that the resulting embedding Γ_ν of $G^-(\nu)$ is proper.

Root nodes: For the root R- or P-node ν, we can obtain a proper embedding Γ of G by replacing each virtual edge e in a proper embedding γ_ν of $\sigma(\nu)$ with a proper embedding Γ_e according to the same rule (β).

This completes an inductive proof for the existence of proper embeddings in canonical instances when there is no forbidden subgraph. It is not difficult see that the above procedure for constructing a proper embedding of G can be implemented to run in linear time. Therefore, this completes the proof of Theorem 6.

References

[1] P. Angelini, M. D. Bartolomeo, and G. D. Battista, Implementing a partitioned 2PB-embedding testing algorithm, In 20th International Symposium on Graph Drawing (GD '12), Springer-Verlag, Lecture Notes in Computer Science, 2012.

[2] F. Bernhart and P. C. Kainen, The book thickness of a graph, J. Combin. Theory Ser. B 27(3), 320-331, 1979.

[3] F. R. K. Chung, F. T. Leighton, and A. L. Rosenberg, Embedding graphs in books: A graph layout problem with applications to VLSI design, SIAM J. Algebraic Discrete Methods, 8, pp. 33-58, 1986.

[4] G. Di Battista and R. Tamassia, On-line planarity testing, SIAM J. on Comput. 25(5), pp. 956-997, 1996.

[5] G. Di Battista and R. Tamassia, On-line maintenance of triconnected components with SPQR-trees, Algorithmica, 15, pp. 302-318, 1996.

[6] V. Dujmović and D. R. Wood, On linear layouts of graphs Discrete Math. Theor. Comput. Sci, 6, pp. 339-358, 2004.

[7] V. Dujmović and D. R. Wood, Stacks, queues and tracks: Layouts of graph subdivisions, Discrete Math. Theor. Comput. Sci, 7, pp. 155-202, 2005.

[8] S. Hong, and H. Nagamochi, Two-page book embedding and clustered graph planarity, TR[2009-004], Dept. of Applied Mathematics and Physics, University of Kyoto, Japan, 2009.

[9] J. E. Hopcroft and R. E. Tarjan, Dividing a graph into triconnected components, SIAM J. on Comput., 2, pp. 135-158, 1973.

[10] A. L. Rosenberg, The Diogenes approach to testable fault-tolerant arrays of processors, IEEE Trans. Comput. C-32, pp. 902-910, 1983.

[11] W. T. Tutte, Graph Theory, Encyclopedia of Mathematics and Its Applications, Vol. 21, Addison-Wesley, Reading, MA, 1984.

[12] M. Yannakakis, Embedding planar graphs in four pages, J. Comput. System Sci. 38, pp. 36-67, 1989.

[13] A. Wigderson, The complexity of the Hamiltonian circuit problem for maximal planar graphs, Technical Report 298, EECS Department, Princeton University, 1982.

Appendix 1: SPQR tree

We review the definition of *triconnected components* [17] (or *3-blocks*) and a variation of the *SPQR tree* [9, 10] of a biconnected graph.

First we review the definition of triconnected components [17]. If G is triconnected, then G itself is the unique triconnected component of G. Otherwise, let u, v be a cut-pair of G. We split the edges of G into two disjoint subsets E_1 and E_2, such that $|E_1| > 1$, $|E_2| > 1$, and the subgraphs G_1 and G_2 induced by E_1 and E_2 only have vertices u and v in common. Form the graph G_1' from G_1 by adding an edge (called a *virtual edge*) between u and v that represents the existence of the other subgraph G_2; similarly form G_2'. We continue the splitting process recursively on G_1' and G_2'. The process stops when each resulting graph reaches one of three forms: a triconnected simple graph, a set of three multiple edges (a triple bond), or a cycle of length three (a triangle).

The triconnected components of G are obtained from these resulting graphs:

- a triconnected simple graph;
- a *bond*, formed by merging the triple bonds into a maximal set of multiple edges;
- a *polygon*, formed by merging the triangles into a maximal simple cycle.

The triconnected components of G are unique. See [17] for further details.

One can define a tree structure, sometimes called as the *3-block tree*, using triconnected components as follows. The nodes of the 3-block tree are the triconnected components of G. The edges of the 3-block tree are defined by the virtual edges, that is, if two triconnected components have a virtual edge in common, then the nodes that represent the two triconnected components in the 3-block tree are joined by an edge that represents the virtual edge.

There are many variants of the 3-block tree in the literature; the first was defined by Tutte [20]. In this paper, we use the terminology of the *SPQR tree*, as defined by di Battista and Tamassia [9, 10]. We now briefly review this terminology.

Each node ν in the SPQR tree is associated with a graph $G = (V, E)$ called the *skeleton* of ν, denoted by $\sigma(\nu) = (V_\nu, E_\nu)$ $(V_\nu \subseteq V)$. For each edge $e = (u, v) \in E_\nu$, G has an induced subgraph G_e which shares $\{u, v\}$ as a cut-pair with the complementary part $G - (V(G_e) - \{u, v\})$. Hence the skeleton $\sigma(\nu)$ corresponds to a triconnected component, providing an abstract structure of the entire graph G. There are four types of nodes in the SPQR tree. The node types and their skeletons are:

1. Q-node: the skeleton consists of two vertices connected by two multiple edges. Each Q-node corresponds to an edge of the original graph.
2. S-node: the skeleton is a simple cycle with at least three vertices (this corresponds to a polygon triconnected component).
3. P-node: the skeleton consists of two vertices connected by at least three edges (this corresponds to a bond triconnected component).
4. R-node: the skeleton is a triconnected graph with at least four vertices.

The SPQR tree as developed by di Battista and Tamassia is a data structure with efficient operations. In this paper, we use the SPQR tree only as a convenient way to traverse the 3-blocks of a biconnected graph. In fact, we use a slight modification of the SPQR tree: we omit the Q-nodes and we root the tree as described below. We will refer the (modified) SPQR tree as the SPQR tree throughout this paper.

In this paper, a given graph G is assumed to be canonical and it has no multiple edges or a pair of red and blue edges with the same end vertices.

The SPQR tree is unique [9, 10]. We treat the SPQR tree of a graph G as a rooted tree $\mathcal{T}$ by choosing an arbitrary node ν^* as its root.

Some further notation for the SPQR tree is required. Suppose that G is a biconnected (but not triconnected) planar graph, and $\mathcal{T}$ is the rooted SPQR tree of G. Let ν be a non-root node in $\mathcal{T}$, and μ be the parent of ν. The graph $\sigma(\mu)$ has one virtual edge e in common with $\sigma(\nu)$. The edge e is the *parent virtual edge* in $\sigma(\nu)$, and it is a *child virtual edge* in $\sigma(\mu)$. For each non-root node v of the SPQR tree, $\sigma(\nu)$ has precisely one parent virtual edge, and for each non-leaf node ν', $\sigma(\nu')$ has at least one child virtual edge. A node ν which is neither the root or a leaf node is called an *internal* node.

We denote the graph formed from $\sigma(\nu)$ by deleting its parent virtual edge as $\sigma^-(\nu)$, if ν is not the root of $\mathcal{T}$. If $\sigma(\nu)$ is a non-root R-node, then $\sigma^-(\nu)$ is internally triconnected. The union of the graphs $\sigma^-(\mu)$ for all descendants μ of ν, including ν itself, is denoted by $G^-(\nu)$; i.e., $G(\nu)$ is the graph obtained from G by inducing the vertex set $\cup\{V_\mu \mid$ descendants μ of $\nu\}$.